

An Automated DAKOTA and VULCAN-CFD Framework with Application to Supersonic Facility Nozzle Flowpath Optimization

Erik L. Axdahl
NASA Langley Research Center, Hampton, Virginia

NASA STI Program ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

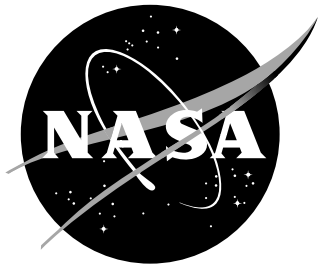
- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI Program, see the following:

- Access the NASA STI program home page at [*http://www.sti.nasa.gov*](http://www.sti.nasa.gov)
- E-mail your question to [*help@sti.nasa.gov*](mailto:help@sti.nasa.gov)
- Phone the NASA STI Help Desk at 757-864-9658
- Write to:
NASA STI Information Desk
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199



An Automated DAKOTA and VULCAN-CFD Framework with Application to Supersonic Facility Nozzle Flowpath Optimization

Erik L. Axdahl

NASA Langley Research Center, Hampton, Virginia

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

August 2015

Acknowledgments

The author wishes to thank Dr. Richard Gaffney for his help with running the Irrotational Method of Characteristics for Nozzle Design code used in the design of the nozzle profiles as well as prior discussions on the philosophy of nozzle design. Additional acknowledgement is made to Dr. Robert Baurle and Mr. Michael Bynum for project planning activities leading up to the execution of this work.

<p>The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.</p>

This report is available in electronic form at
<http://ntrs.nasa.gov>

Abstract

Removing human interaction from design processes by using automation may lead to gains in both productivity and design precision. This memorandum describes efforts to incorporate high fidelity numerical analysis tools into an automated framework and applying that framework to applications of practical interest. The purpose of this effort was to integrate VULCAN-CFD into an automated, DAKOTA-enabled framework with a proof-of-concept application being the optimization of supersonic test facility nozzles. It was shown that the optimization framework could be deployed on a high performance computing cluster with the flow of information handled effectively to guide the optimization process. Furthermore, the application of the framework to supersonic test facility nozzle flowpath design and optimization was demonstrated using multiple optimization algorithms.

Contents

1 Nomenclature	2
2 Introduction	4
3 Methodology	4
3.1 Computational Tools	4
3.2 High Performance Computing Job Structure	7
3.3 Reference Conditions and Requirements	8
3.4 Optimization	8
4 Results and Discussion	10
5 Summary and Conclusions	11
6 Directions of Future Work	13
A DAKOTA Driver Script	16
B DAKOTA Input Files	18
C Optimization Problem Definition	21
D Representative VULCAN-CFD Solution	23
E DAKOTA Iteration Data	28
E.1 DIRECT Method	28
E.2 COBYLA Method	30
E.3 Quasi-Newton Method	31

1 Nomenclature

Acronyms/ Initialisms	Description
AHSTF	A rc- H eated S cramjet T est F acility
BFGS	B royden, F letcher, G oldfarb, S hanno
CFD	computational fluid dynamics
CFL	C ourant, F reidrichs, L ewy
COBYLA	constrained optimization b y linear approximation
DAKOTA	d esign a nalysis k it for optimization and t erascale applications
DIRECT	d ividing r ectangles
DAF	d iagonal approximate factorization
HPC	h igh performance computing
IMOCND	irrotational m ethod of characteristics for n ozzle d esign
LDFSS	low dissipation flux vector split scheme
MOC	m ethod of characteristics
MUSCL	m onotonic upstream-centered scheme for conservation laws
NASA	N ational A eronautics and S pace A dministration
PBS	p ortable b atch system
VULCAN	viscous upwind algorithm for complex flow analysis

Latin Symbols	Description	Units
A_{core}	nozzle exit core area	m^2
\tilde{A}_{core}	normalized nozzle exit core area	1
A_{exit}	Nozzle exit area	m^2
a_i	weighting coefficient	1
e	error	1
F	objective function	1
\tilde{F}	pseudo-objective function	1
f_{eval}	function evaluations	1
H_{exit}	nozzle exit height	m
i	number of algorithm iterations	1
L_{nozzle}	nozzle length	m
\tilde{L}_{nozzle}	nondimensional nozzle length	1
L_{ref}	reference nozzle length	m
M	Mach number	1
$M_{centerline}$	nozzle exit centerline Mach number	1
$\tilde{M}_{centerline}$	scaled nozzle exit centerline Mach number	1
M_{design}	targeted nozzle exit design Mach number	1
M_{MOC}	nozzle exit Mach number from Method of Characteristics	1
\tilde{M}_{MOC}	scaled nozzle exit Mach number from Method of Characteristics	1
P	penalty function	1
\tilde{p}	scaled pressure	1
p_0	stagnation pressure	Pa

\tilde{p}_0	scaled stagnation pressure	1
Q	number of processors required per function evaluation	1
R	penalty parameter	1
R_c	nozzle throat radius of curvature	m
\tilde{R}_c	nondimensional nozzle throat radius of curvature	1
R_t	nozzle throat half-length from centerline	m
S	number of processors available	1
T	static temperature	K
\tilde{T}	scaled static temperature	1
T_0	stagnation temperature	K
X, Y, Z	cartesian coordinates	m
y^+	nondimensional distance to the wall	1

Greek **Description**

Symbols

ϕ variable placeholder

Superscripts **Description**

$*$ optimized condition

2 Introduction

Efforts have been made in the Hypersonic Airbreathing Propulsion Branch at NASA Langley Research Center to improve the automation of design and analysis tools and methods used at multiple levels of fidelity ranging from engine cycle analysis to high fidelity CFD. By gradually removing the human from the design loop, productivity can be increased by exploring larger design spaces and achieving designs that are closer to the global optimum or that possess improved robustness. Furthermore, automating the tools used in the design process will ease studies that provide quantification of uncertainty and sensitivity. This will allow for improved design knowledge for decision making by project leaders, designers, and analysts alike.

The design of facility nozzles is a good application area for tool automation because the design process can be generalized to an automated framework. The method of characteristics (MOC) is a method by which one may design a nozzle wall contour for a given exit Mach number, but because the assumptions underlying the method do not account for viscosity, boundary layer buildup in the nozzle flow will produce both vorticity due to boundary layer viscous effects as well as an exit Mach number that is off design because the boundary layer inhibits expansion of the flow by reducing the effective cross-sectional area. Additionally, the presence of the boundary layer has an adverse impact on exit flow uniformity. Therefore, the designer would typically be required to adjust the nozzle design using MOC manually until the desired exit Mach number is achieved in a viscous solution within some specified tolerance.

The purpose of the effort described in this memorandum was to demonstrate a capability for integrating the high-fidelity VULCAN-CFD code in an automated framework driven by the DAKOTA toolkit. To that end, a proof-of-concept supersonic facility nozzle optimization study was initiated to demonstrate the practical methodology of executing the framework on a high performance computing cluster. Three optimization algorithms were chosen in order to demonstrate the automated framework varying levels of algorithmic fidelity and efficiency. At the conclusion of the effort, a capability was gained in automated supersonic facility nozzle optimization, which may provide a starting point for problems in other research areas requiring a similar automated framework.

3 Methodology

3.1 Computational Tools

Computer codes used in an automated framework require the capability to have variables passed to them without direct human interaction. Therefore, each code must have the ability to gather its inputs from a text-based input file, via options on the command line, an application programming interface, or a combination of all three. In order to adjust design variables and simulation parameters appearing in input files, template input files for each code were defined with placeholder strings occupying locations in each template where the framework driver may substitute a variable value. At the conclusion of the sequence of codes to be run, numerical values

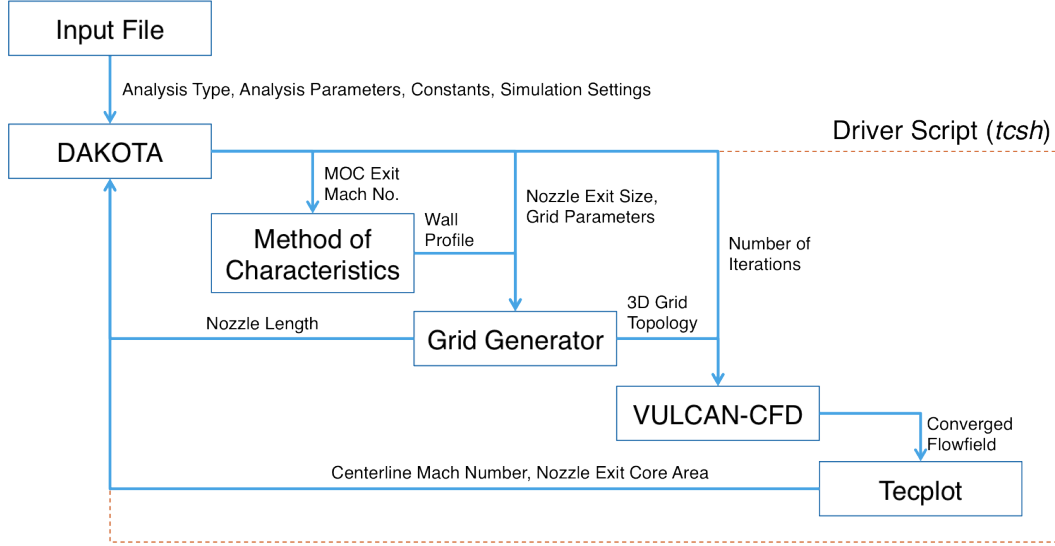


Figure 1. Flow of information for the described simulation framework.

representing the final results were extracted and placed in a results file that was read by the framework driver. For the proof-of-concept supersonic facility nozzle design problem, codes were run sequentially in order to define the nozzle profile, generate the three-dimensional numerical grid, run the CFD analysis, and extract the results that were fed back to the framework driver in order to record results and assign values to the design variables for subsequent iterations. The diagram in fig. 1 maps to the following discussion in order to visualize the flow of information from one code to the next.

Non-dimensionalized, two-dimensional nozzle profiles were generated using a code written by Richard Gaffney of the Hypersonic Airbreathing Propulsion Branch at NASA Langley Research Center. The IMOCND code¹ uses an MOC procedure to design an inviscid nozzle profile given input parameters such as target nozzle exit Mach number and non-dimensional throat radius of curvature. Because the method produces non-dimensionalized nozzle profiles, the resulting nozzle wall contour was scaled such that the exit height matched the design requirement.

Final nozzle wall profile definition and grid generation were completed via a custom script that took as input the nozzle wall profile from the IMOCND code and produced a three dimensional computational grid along the entire nozzle length from the subsonic inflow to the supersonic exit. The profile was scaled given a user-defined exit size and the subsonic nozzle contour was curve fit to the subsonic entry point in order to achieve a fixed nozzle entrance wall angle. Grid spacing was defined by a user-specified number of longitudinal and transverse points as well as clustering parameters near the walls and near the throat.² Clustering was enforced to achieve sufficient resolution in the wall boundary layer as well as in areas where rapid area changes occur.

The nozzle flow properties were simulated using the VULCAN-CFD^{3,4} code, developed and maintained at NASA Langley Research Center. VULCAN-CFD uses a

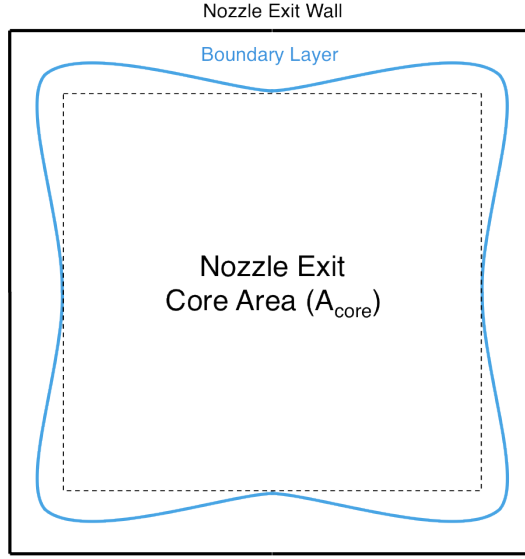


Figure 2. A notional schematic is shown of the definition of the core flow area in the nozzle exit. The core area is defined here as the largest rectangular area that fits completely within the region of flow outside of the boundary layer.

finite-volume, cell-centered scheme for solving flows on structured grids. The nozzle flow for this study was assumed to be viscous, thermally perfect, and chemically frozen with non-vitiated air. The flow fluxes were computed using LDFSS⁵ and were recombined with a 3rd order MUSCL interpolation strategy. The flow was integrated temporally using the ILU scheme⁶ with local time stepping and an increasing CFL number schedule. Turbulence was modeled using the two-equation Menter $k-\omega$ model⁷ with a turbulent Prandtl number of 0.9. Wall matching functions⁸ were used for wall-bounded areas of the solution domain to relax boundary layer grid resolution requirements. Finally, a coarse, medium, and fine grid sequencing technique was used to speed up the time required for solution convergence.

Solution quantities of interest were extracted from the VULCAN-CFD flow field using Tecplot*. At the conclusion of each simulation, the exit plane data from the nozzle was extracted and exported to a data file that was read and processed by a custom script. Regions in the plane where the exit flow Mach number were within 1% of the design Mach number were identified so that the core size could be determined as an engineering estimate. A notional schematic of the definition of the nozzle core is shown in fig. 2. The ratio of the core size to the nozzle exit size was used to constrain the optimization process with an ideal value of unity. The centerline Mach number was also extracted to drive the optimization process.

The codes used in the nozzle profile definition, grid generation, flow simulation, and parameter extraction were integrated using the Sandia National Laboratory-developed and -maintained code DAKOTA.⁹ DAKOTA is a toolkit for integrating codes in order to conduct higher order analyses such as optimization, uncertainty

*Tecplot is a trademark of Tecplot, Inc.

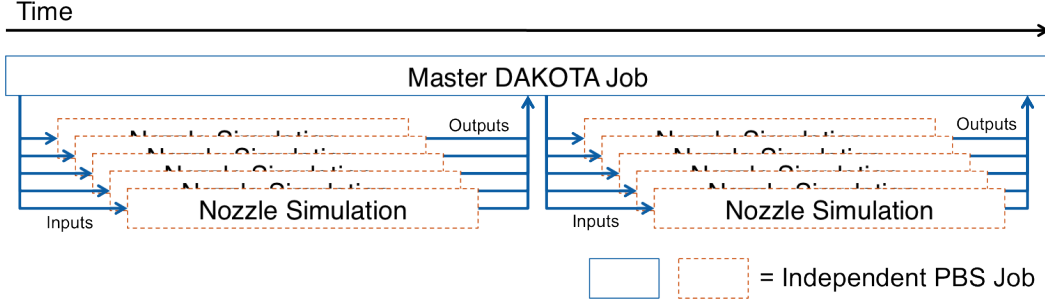


Figure 3. A schematic of the high performance computing architecture is shown. A master DAKOTA instance is run from its own, long-running PBS job while spawning independent nozzle simulations on their own PBS jobs with unique design variables.

quantification, parameter estimation, and design space exploration. The flow of information from one code to the next was controlled in a black-box manner using shell scripts, in this case written in *tcs*h, and analysis outputs were fed back into DAKOTA in order to generate new sets of design variable values for subsequent simulation runs. Appendix A shows an example of the shell script used to drive the analysis in this work. A powerful aspect of DAKOTA is that different types of analyses may be run simply by altering the master input file fed to DAKOTA without making any further changes to the driver scripts controlling the execution of codes, flow of information, and parameter extraction in the analysis framework. Appendix B shows an example of a DAKOTA input file used in this work.

3.2 High Performance Computing Job Structure

Because design iterations were carried out on an HPC cluster with multiple simulations carried out simultaneously and independently of each other, attention was paid toward the practical methodology of running the automated framework described in section 3.1. Table 1 summarizes candidate methods for how to run DAKOTA and the simulations it drives depending on whether each are run in a serial or parallel mode and the type of HPC resources and settings available to the user. As shown schematically in fig. 3, a job architecture was chosen corresponding to Case 4 consisting of a master DAKOTA instance on a long-duration PBS job spawning independent simulations in PBS jobs each with unique sets of design variable values. While Case 3 may also be suitable for the present analysis, such a method requires techniques such as processor tiling as well as possibly wasting job resources if they are not all needed the entire time. Considering this, the ability to run many, smaller PBS jobs asynchronously allowed individual simulations to be run as resources were freed on the shared HPC resource. Furthermore, submitting jobs separately allowed cluster resources to be used as needed as the number of simultaneous simulations required do not necessarily remain constant during the optimization process.

Table 1. DAKOTA and application parallelism use cases (table reproduced from information in DAKOTA user manual⁹ and associated presentation packaged with source code)

Case	DAKOTA	Simulation	Notes
1	Parallel	Serial	$S - 1$ (or S if DAKOTA run on login node) simultaneous simulation instances, each requiring $Q = 1$ processors
2	Serial	Parallel	One simultaneous simulation instance at a time on Q processors
3	Serial	Parallel	Run $\approx (S - 1)/Q$ or $\approx S/Q$ simultaneous, Q processor simulations on a single job using processor tiling
4	Serial	Parallel	Run DAKOTA on a login node or its own job and submit simulation jobs each with Q processors to a scheduler (e.g., <i>qsub</i>)

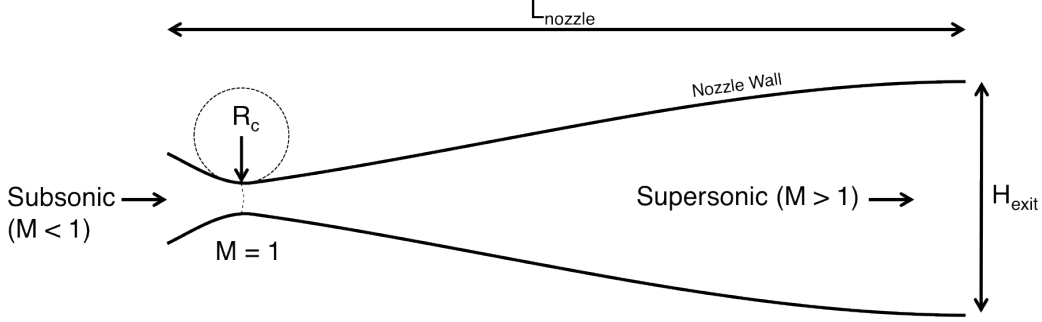


Figure 4. A notional schematic is shown of the nozzle, including the definition of the nozzle length and throat radius of curvature.

3.3 Reference Conditions and Requirements

The nozzle design requirements were selected in order to reflect those of a previously exercised, manual design activity for designing square nozzles for the NASA Langley AHSTF supersonic tunnel.^{10,11,12} For this study a nozzle was designed with a specific target exit Mach number, although the same procedure could be applied to a nozzle design of any given exit Mach number. Note that while vitiated air would normally be present in the gas flowing from the facility plenum,^{13,14} the air composition was assumed to be non-vitiated in the present numerical analyses.

3.4 Optimization

For a nozzle constrained by design exit Mach number and exit size, notionally shown in fig. 4, the principal parameters defining its design when using the IMOCND code are the throat radius of curvature and the MOC exit Mach number. When considering the need to produce a nozzle exit flow profile that is as uniform as

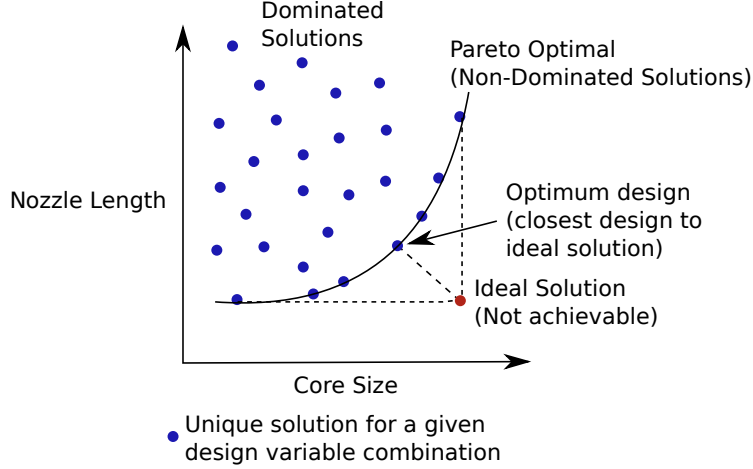


Figure 5. A schematic is shown of notional design space solutions, including the Pareto front and the definition of the ideal solution.

possible, increasing the throat radius of curvature is one way to accomplish this as large radii produce a sonic line at the throat that approaches a linear shape. Furthermore, large throat radii of curvature cause a slower expansion with reduced flow gradients. However, increasing the radius of curvature also increases the length of the nozzle which can have adverse effects on the manufacturability of the nozzle as well as what the facility itself can accommodate. In addition, the boundary layer will become thicker with longer nozzles,¹⁵ which impacts exit core size and flow uniformity. Therefore the optimization problem has two competing objectives of nozzle length and exit flow uniformity (surrogated here as the nozzle exit core size) with the centerline Mach number being an equality constraint. The exit centerline Mach number was chosen as a constraint instead of part of the objective function because there are multiple nozzle solutions that may satisfy that constraint. The multi-objective function to be minimized was expressed as

$$F(\tilde{R}_c, \tilde{M}_{MOC}) = -a_1 \tilde{A}_{core} + a_2 \tilde{L}_{nozzle} \quad (1)$$

where the weighting coefficients a_1 and a_2 may take on any values such that they add to unity. The values of a in a multi-objective function are generally defined such that $a_i \in [0, 1]$ and $\sum_i a_i = 1$. Because the objective function in this case is to be minimized, the coefficient of a_1 is negative due to the need to maximize the core area and the coefficient of a_2 is positive due to the need to minimize nozzle length. For this study, an equal weighting to both objectives was chosen such that $a_1 = a_2 = 0.5$. For any given value of a_1 and a_2 , the optimum value of the objective function F yields a design on the Pareto optimal front of the design space, shown notionally in fig. 5, which allows design trades to be done using different objective function weightings. In this study, the core area and nozzle length were non-dimensionalized in order to make them the same order of magnitude. Therefore, $\tilde{A}_{core} = A_{core}/A_{exit}$ and $\tilde{L}_{nozzle} = L_{nozzle}/L_{ref}$. For more information on the mathematical optimization definition in this study, refer to appendix C.

Of the optimization methods available in DAKOTA, three were chosen to demonstrate the capability of the automated framework. The first was the DIRECT¹⁶ method, which is a zeroth order method designed for engineering applications that scales well with the number of design variables in the parameter space. DIRECT is also guaranteed to find the global minimum within a user-defined spatial tolerance. The number of simultaneous design points the method may evaluate is dependent both on the number of regions where potential global minima have been identified as well as the dimensionality of the problem. The second was the COBYLA^{17,18} method, which traverses the design space by producing successive linear approximations of the objective function and constraints separately in an $n + 1$ dimensional simplex. The COBYLA method may only evaluate a single design point at a time and converges according to a fixed step reduction schedule. The third was a Quasi-Newton method⁹ that is distinct from a classical Newton method in that the approximate inverse Hessian at a point is updated using the BFGS^{19,20,21,22} method using the local Jacobian. This makes the method quasi-second order where local Jacobians are computed numerically—rather than analytically—in this study by perturbing the locally sampled point at each iteration. As the calculation of numerical Jacobians increases the computational resources required by the algorithm also increases, although a CFD code that incorporates an adjoint capability may be able to mitigate this. As with the COBYLA method, the Quasi-Newton method may only evaluate a single design point at a time, although numerical Jacobians may be computed concurrently at the design point. Furthermore, a vulnerability of both the COBYLA and Quasi-Newton methods is that they may converge to a local minima instead of the global minima.

4 Results and Discussion

The optimization algorithms described in section 3.4 were run to convergence to the constrained optimum design variable values. For more information on numerical results and analysis using VULCAN-CFD, refer to appendix D for a run at the optimum location as determined by the Quasi-Newton method that is representative of each iteration of the process. A list of objective function values for each algorithm at each iteration are summarized in appendix E. Optimum designs as determined by each optimization algorithm are summarized in table 2. The number of function evaluations represents the number of simulations required to converge the algorithm. The number of iterations represents the number of candidate design points evaluated in the convergence of each algorithm. For each algorithm’s converged optimum, the number of function evaluations and total number of algorithmic iterations shows that the Quasi-Newton method is the most efficient of the three. The reason is because the method takes both numerical Jacobians and approximate inverse Hessians into account, thereby improving the convergence characteristics of the method over zeroth and first order methods. While the COBYLA and Quasi-Newton methods both converge to approximately the same design point, the Quasi-Newton method is more efficient in terms of function evaluations by over a factor of two. Another notable aspect of these results is that the DIRECT method converged to the shorter

Table 2. Results for different optimization algorithms are shown. Design variables with asterisks indicate the optimum design variable combination. The listed Mach numbers are scaled against the nozzle design Mach number.

Algorithm	\tilde{R}_c^*	\tilde{M}_{MOC}^*	$\tilde{M}_{centerline}$	$\tilde{A}_{core} [m^2]$	$\tilde{L}_{nozzle} [m]$	f_{eval}	i
DIRECT	5.278	1.039	1.002	0.778	0.278	73	73
COBYLA	7.014	1.040	1.012	0.778	0.293	44	44
Quasi-Newton	7.105	1.041	1.001	0.778	0.290	21	7

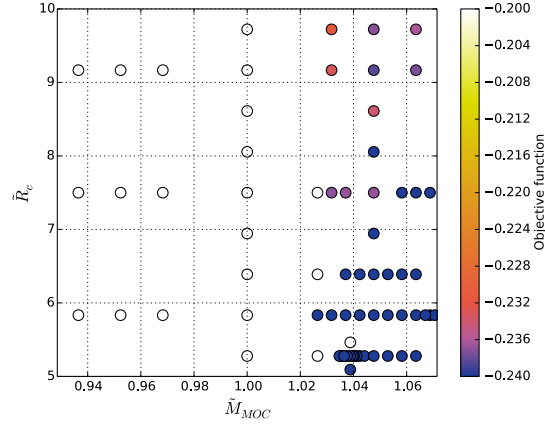
length nozzle while achieving nozzle core area and exit centerline Mach number similar to those determined using the Quasi-Newton method. This suggests that the COBYLA and Quasi-Newton methods converged into a local minima while the DIRECT method identified the location of the global optimum.

Graphical visualizations of each algorithm’s design space exploration are shown in fig. 6. While the sequence of iterations should not be inferred by any of the plots, each gives indications of the methodology and the efficiency of each method. As shown in fig. 6(a), the DIRECT method sampled the design space at gradually refining resolution, probing areas where potential global minima were located. The method terminated once a defined search size tolerance was reached. The pattern of the search indicates that the region where COBYLA and Quasi-Newton methods converged was investigated by the DIRECT algorithm as potentially containing a global minimum. The COBYLA algorithm’s search points are shown in fig. 6(b). The method gradually approached the local optimum point by linearizing the design space at successive points. The Quasi-Newton method search points are shown in fig. 6(c). Here the method rapidly approached the local optimum design point through the computation of the numerical Jacobian and approximation of the inverse Hessian at each point.

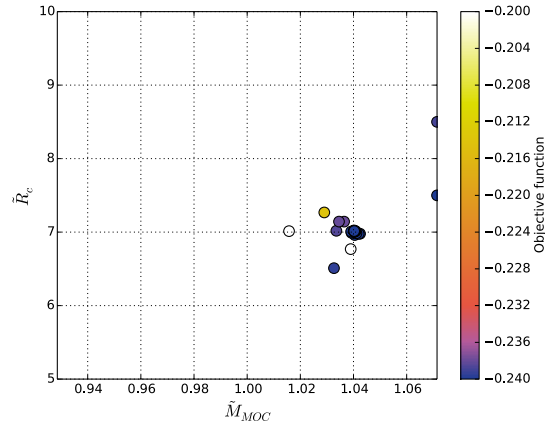
As shown in these results, the Quasi-Newton method has difficulty identifying global minima when local minima are present or if the design space is not smooth or continuous globally. For such problems, the converged solution will be sensitive to the selection of the initial search point. This issue is not problematic with a method such as DIRECT, which guarantees that a global minimum will be found within some user-defined size tolerance. However, this comes at the cost of additional runs relative to higher order methods such as Quasi-Newton. Because the DIRECT algorithm scales well with the number of design variables, a sequential optimization method may be employed where the DIRECT method is allowed to end in the general vicinity of the global minimum with the ending point of DIRECT being the starting point for a Quasi-Newton optimization method. To this end, DAKOTA is able to handle sequential optimization with an arbitrary number of algorithms.

5 Summary and Conclusions

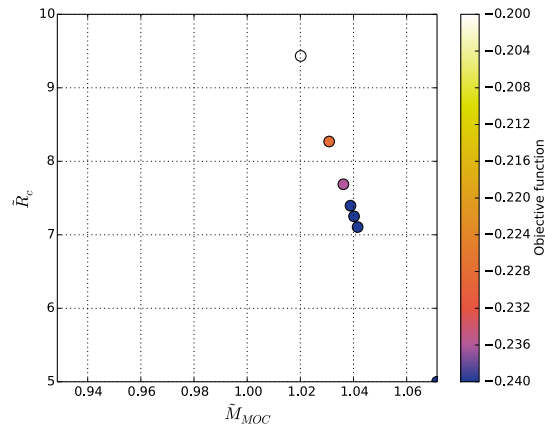
A method for integrating the high-fidelity VULCAN-CFD code in an automated DAKOTA-driven framework was demonstrated with application to the design optimization of a supersonic facility nozzle at conditions representative of testing in



(a)



(b)



(c)

Figure 6. Design space exploration points using the (a) DIRECT, (b) COBYLA, and (c) Quasi-Newton methods. Points indicate locations of iterations and colors indicate the pseudo-objective function value at that point. The color map is saturated above -0.2 and below -0.24.

the NASA Langley AHSTF. The optimization framework was demonstrated for the same facility stagnation conditions and design requirements but with different optimization methods demonstrating zeroth and quasi-second order sampling characteristics. The benefit of utilizing this framework for facility nozzle design is in the automated optimization without user input or decision making until the process has completed, thereby speeding up the design process relative to manual processes for the same level of design precision. Using the same framework, objective trades can be done by selecting different values of the objective weightings in eq. (1) and producing a Pareto front of designs. Once a suite of optimized designs are produced, a single design may be selected by considering facility or manufacturing constraints and maximizing for a particular objective such as nozzle core area. Alternatively, the designer may choose a design on the Pareto front that is closest to the ideal design according to the minimum L_2 -norm distance to the best possible objective function values of all observed optima. This is depicted in fig. 5.

6 Directions of Future Work

The purpose of this study was to demonstrate the value added by integrating the high-fidelity VULCAN-CFD code in an automated framework driven by DAKOTA. For the present application in multi-objective optimization, further improvements can be made to the methodology to achieve better nozzle designs. For example, instead of using the centerline Mach number, the mean core Mach number could be automatically extracted and used instead. Using that value of Mach number would provide a more integrated estimate of the core Mach number instead of using a single value. Furthermore, constraints may be applied to the statistical characterizations of core flow quantities during the optimization process. This would allow the designer to place upper bounds on the allowable departures from flow uniformity during the optimization process according to accepted design criteria or requirements.

While the supersonic facility nozzle optimization problem was chosen as a proof-of-concept due to the well-defined nature of the design space and the relatively few number of design variables affecting aerodynamic nozzle performance, elements of the present framework may be applied to other problems of interest. The same framework could be used to conduct studies in uncertainty quantification and sensitivity analyses to design more robust nozzles and characterize the effect of uncertainties in tunnel plenum conditions. Uncertainties due to manufacturing could also be identified, characterized, and integrated into the analysis. Furthermore, the same HPC job structure and modification to some of the driver scripts will allow the VULCAN-CFD and DAKOTA framework to be applied to other research areas of relevance to high speed flowpaths.

References

- [1] R. L. Gaffney, B. K. Stewart, and S. F. Harvin. “The Design of a High-Q, Mach-5 Nozzle for the NASA Langley 8-Foot HTT”. In: *25th Aerodynamic Measurement Technology and Ground Testing Conference*. AIAA-2006-2954. San Francisco, CA, June 2006.
- [2] K. A. Hoffmann and S. T. Chiang. *Computational Fluid Dynamics*. Fourth. Vol. 1. Engineering Education System, 2000. ISBN: 0962373109. URL: <http://www.worldcat.org/isbn/0962373109>.
- [3] J. White and J. Morrison. “A Pseudo-Temporal Multi-Grid Relaxation Scheme for Solving the Parabolized Navier-Stokes Equations”. In: *14th Computational Fluid Dynamics Conference*. AIAA-1999-3360. Norfolk, VA, Nov. 1999.
- [4] NASA Langley Research Center. *VULCAN-CFD Ver. 6.3*. Miscellaneous. Hampton, VA, June 2014. URL: <http://vulcan-cfd.larc.nasa.gov>.
- [5] J. R. Edwards. “A Low-Diffusion Flux-Splitting Scheme for Navier-Stokes Calculations”. In: *Computers & Fluids* 26.6 (1997), pp. 635–659.
- [6] D. Litton, J. Edwards, and J. White. “Algorithmic Enhancements to the VULCAN Navier-Stokes Solver”. In: *16th Computational Fluid Dynamics Conference*. AIAA-2003-3979. Orlando, FL, June 2003.
- [7] F. R. Menter. “Zonal Two Equation k - ω Turbulence Models for Aerodynamic Flows”. In: *24th Fluid Dynamics Conference*. AIAA-1993-2906. Orlando, FL, July 1993.
- [8] D. Wilcox. “Wall Matching, a Rational Alternative to Wall Functions”. In: *27th Aerospace Sciences Meeting*. AIAA-1989-0611. Reno, NV, Jan. 1989.
- [9] B. Adams, L. Bauman, W. Bohnhoff, K. Dalbey, M. Ebeida, J. Eddy, M. Eldred, P. Hough, K. Hu, J. Jakeman, L. Swiler, and D. Vigil. *DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 5.4 Theory Manual*. Technical Report SAND2011-9106. Updated 11 2013. Sandia National Laboratories, Dec. 2011.
- [10] S. Thomas and R. Guy. “Increased Capabilities of the Langley Mach 7 Scramjet Test Facility”. In: *18th Joint Propulsion Conference*. AIAA-1982-1240. Cleveland, OH, June 1982.
- [11] S. R. Thomas and R. W. Guy. *Expanded Operation Capabilities of the Langley Mach 7 Scramjet Test Facility*. Technical Paper 2186. NASA, Oct. 1983.
- [12] S. R. Thomas, R. T. Volland, and R. W. Guy. “Test Flow Calibration Study of the Langley Arc-Heated Scramjet Test Facility”. In: *23rd Joint Propulsion Conference*. AIAA-1987-2165. San Diego, CA, 1987.
- [13] W. MacDermott, D. Horn, and C. Fischer. “Flow Contamination and Flow Quality in Arc Heaters used for Hypersonic Testing”. In: *17th Aerospace Ground Testing Conference*. AIAA-1992-4028. Nashville, TN, July 1992.

- [14] K. Fischer and K. Rock. “Calculated Effects of Nitric Oxide Flow Contamination on Scramjet Performance”. In: *31st Joint Propulsion Conference and Exhibit*. AIAA-1995-2524. San Diego, CA, July 1995.
- [15] J. Crown and W. Heybey. *Supersonic Nozzle Design*. Naval Ordnance Laboratory Memorandum 10594. U.S. Naval Ordnance Laboratory, Apr. 1950.
- [16] D. Jones, C. Perttunen, and B. Stuckman. “Lipschitzian Optimization Without the Lipschitz Constant”. In: *Journal of Optimization Theory and Application* 79.1 (Oct. 1993), pp. 157–181.
- [17] M. Powell. “Advances in Optimization and Numerical Analysis”. In: ed. by S. Gomez and J.-P. Hennart. Kluwer Academic, 1994. Chap. A Direct Search Optimization Method that Models the Objective and Constraint Functions by Linear Interpolation, pp. 51–67.
- [18] M. Powell. *A View of Algorithms for Optimization without Derivatives*. Apr. 2007.
- [19] C. Broyden. “The Convergence of a Class of Double Rank Minimization Algorithms, Parts I and II”. In: *Journal of the Institute of Mathematics and its Applications* 6 (1970), pp. 76–90, 222–231.
- [20] R. Fletcher. “A New Approach to Variable Metric Algorithms”. In: *The Computer Journal* 13 (1970), pp. 317–322.
- [21] D. Goldfarb. “A Family of Variable Metric Methods Derived by Variational Means”. In: *Mathematics of Computation* 24 (1970), pp. 23–36.
- [22] D. Shanno. “Conditioning of Quasi-Newton Methods for Function Minimization”. In: *Mathematics of Computation* 24 (1970), pp. 647–656.
- [23] G. N. Vanderplaats. *Numerical Optimization Techniques for Engineering Design with Applications*. McGraw-Hill, 1984.
- [24] S. P. Schneider. *Method of Characteristics Design of a Supersonic Wind Tunnel Nozzle with Square Cross-Section*. Contractor Report 194359. NASA, Jan. 1993.

Appendix A

DAKOTA Driver Script

A sample *tcs* driver script controlling the execution of codes and flow of information in DAKOTA is shown.

```
#!/bin/tcsh
# See Advanced Simulation Code Interfaces chapter in Users Manual
# $argv[1] is params.in from Dakota
# $argv[2] is results.out returned to Dakota

# Extract run number
set num=`pwd | awk -F. '{print $NF}`

# Create VULCAN Input File
../../scripts/dprepro $argv[1] \
    nozzle_unsplit_real_template.inp \
    nozzle_unsplit_real.inp
rm nozzle_unsplit_real_template.inp

# Create MOC Input File
../../scripts/dprepro $argv[1] \
    input_template.dat \
    input.dat
echo "Generated MOC Input File"

# Create postprocess script
../../scripts/dprepro $argv[1] \
    ../../scripts/outflow_reduce_template.py \
    ../../scripts/outflow_reduce.py
echo "Generated post-processing script"

../../scripts/dprepro $argv[1] \
    ../../scripts/generate_grid_template \
    ../../scripts/generate_grid
echo "Generated grid generation script"

# Create PBS Script
sed 's/jobid/$num/g' runfile_template > runfile
rm runfile_template
chmod +x runfile

# Generate wall profile, grid, and split grid and input file
tcsh ../../scripts/generate_grid > grid_generation_output.out
```

```
# Submit job to PBS
qsub runfile

# Wait for job to complete by waiting for file "jobdone"
bash ../../scripts/waitforit

# Move to results directory and copy tecplot files
cd Reomp_files/Plot3d_files
cp ../../../../tecplot/* .

# Extract objective function data
tec360 -b tec_macro.mcr > tecplot_extract.out
python ../../../../scripts/outflow_reduce.py > objective.dat

# Copy objective function value to root
cp objective.dat ../../results.out
```

Appendix B

DAKOTA Input Files

The following is an example DAKOTA input file used for each optimization method outlined in this paper. Bracketed items are placeholders where a user would input a numerical value in the actual input file.

```
# DAKOTA INPUT FILE - dakota.in

strategy,
    tabular_graphics_data
    pareto_set
        method_pointer = 'QNwt'
        weight_sets = 0.5 0.5

method,
    id_method = 'QNwt'
    optpp_q_newton

method,
    id_method = 'C_DIRECT'
    coliny_direct
    min_boxsize_limit = 0.5 # 0.5 converges roughly
    constraint_penalty = 1.0 # Default is 1000.0

method,
    id_method = 'COBYLA'
    coliny_cobyla

model,
    single

variables,
    # MOCMach = The design Mach number for the MOC code
    #             when generating a new nozzle profile. [1]
    # ThroatROC = The throat radius of curvature normalized
    #             against throat height when creating new
    #             designs. [1]
    # M_design = Nozzle design Mach number [1]
    # exit_size = half-side width/height for square
    #             nozzle [inches]
    # M = Longitudinal number of grid points [1]
    # N = Transverse number of grid points [1]
```

```

continuous_design = 2
    descriptors    'MOCMach'    'ThroatROC'
    initial_point  [MOCO]      7.5
    lower_bounds   [MOC1]      5.0
    upper_bounds   [MOCu]      10.0

discrete_state_set_real = 5
    descriptors    'M_design' 'exit_size' 'p0'      'rho0' 'vi'
    set_values      [M_design] [exit_size] [p0]      [rho0] [vi]

discrete_state_set_integer = 5
    descriptors    'M'        'N'        'ITC'      'ITM'      'ITF'
    set_values      257        57         3000       3000       3000

interface,
    fork
    asynchronous
    evaluation_concurrency = 5

    analysis_driver = 'simulatorScript'

    work_directory named 'cases/workdir'
    directory_tag
    directory_save

    parameters_file = 'io/params.in'

    results_file     = 'io/results.out'

    template_directory 'template'

    # Tag files
    file_tag

    # Save files
    file_save

responses,
    objective_functions = 2
    nonlinear_equality_constraints = 1

    descriptors "norm coreArea [1]"
                "norm nozzleLength [1]"

```

```
"norm exitMach [1]"

# Optimization direction for objective functions
sense "max" "min"

# Nonlinear equality constraint target value (Mcenter/Mdesired)
targets 1.0

numerical_gradients
no_hessians
```


Appendix C

Optimization Problem Definition

The constrained optimization problem in this study was defined mathematically as

Minimize:

$$F(\tilde{R}_c, \tilde{M}_{MOC}) = -a_1 \tilde{A}_{core} + a_2 \tilde{L}_{nozzle} \quad (C1a)$$

Subject to:

$$\tilde{M}_{centerline} = 1 \quad (C1b)$$

$$5 \leq \tilde{R}_c \leq 10 \quad (C1c)$$

$$0.93 \leq \tilde{M}_{MOC} \leq 1.07 \quad (C1d)$$

Equation (C1a) is the objective function, which is a linear combination of the response variables of normalized core area and non-dimensionalized nozzle length, which are themselves functions of non-dimensionalized nozzle throat radius and the MOC nozzle exit Mach number. Equation (C1b) is an equality constraint that defines the feasibility of designs as the optimization procedure advances. Here, the constraint requires that designs have an exit Mach number as close to the target design value as possible. Finally, eqs. (C1c) and (C1d) are side constraints that come into play for optimization algorithms that require them such as DIRECT. The bounds are defined such that certain optimization algorithms will only search within a domain that satisfies the side constraints.

While some optimization algorithms handle equality and inequality constraints directly and separately from the objective function (e.g., COBYLA), the DAKOTA framework may alternatively use a pseudo-objective function that incorporates both the objective function and a penalty function that handles the constraints indirectly. For the optimization problem in this study, the pseudo-objective function takes the form

$$\tilde{F} = F(\tilde{R}_c, \tilde{M}_{MOC}) + P(\tilde{M}_{centerline}) \quad (C2)$$

The penalty function itself takes different forms depending on whether equality or inequality constraints are included, but for this study it has the form

$$P(\tilde{M}_{centerline}) = R \left[\tilde{M}_{centerline} - 1 \right]^2 \quad (C3)$$

The form of the penalty function ensures that its minimum is located at the equality constraint value. The penalty parameter R defines the strength of the penalty function and was set to 1.0 in this study. The penalty parameter was kept at this value in order to keep the penalty function value at the same order of magnitude as the other objectives. For more information on the use and definition of penalty functions, refer to Vanderplaats.²³ Hence, the mathematical definition of the optimization problem using penalty functions takes the form

Minimize:

$$\tilde{F} = F(\tilde{R}_c, \tilde{M}_{MOC}) + P(\tilde{M}_{centerline}) \quad (\text{C4a})$$

Subject to:

$$5 \leq \tilde{R}_c \leq 10 \quad (\text{C4b})$$

$$0.93 \leq \tilde{M}_{MOC} \leq 1.07 \quad (\text{C4c})$$

with F and P respectively defined by eqs. (C1a) and (C3).

Appendix D

Representative VULCAN-CFD Solution

The discussion of the CFD solution in this appendix is representative of any given solution during the optimization process. While the nozzle profiles will change from iteration to iteration of the optimization algorithm, the methodology of grid generation, boundary conditions definition, and iteration to convergence is essentially the same. The representative solution chosen here was that of the optimum point as converged to by the Quasi-Newton method in table 2. For more information on the VULCAN-CFD settings used to arrive at this solution, refer to section 3.1.

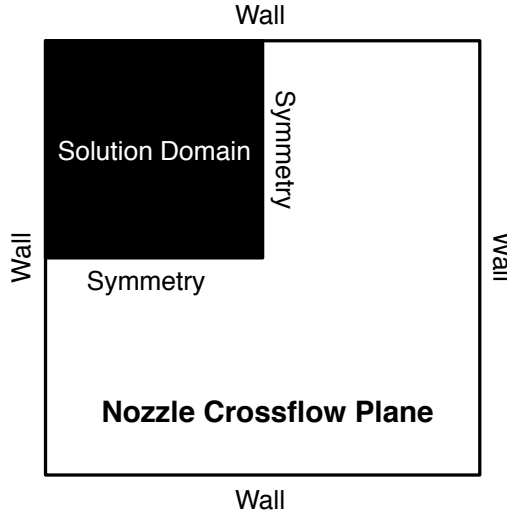


Figure D1. A two-dimensional nozzle area cross-section showing the solution domain as well as wall and symmetry boundary conditions.

The CFD solution of the supersonic nozzle was carried out in a three dimensional simulation representing a quarter of the nozzle domain. This was because the nozzle design has two planes of symmetry in the vertical and lateral axis directions emanating from the center of each cross-flow plane moving downstream. Because of this, two boundaries of the domain were defined to be the adiabatic walls of the nozzle and the other two were defined to be symmetry boundary conditions. This is shown schematically in fig. D1. The dimensions of the grid shown in fig. D2 has 257 points along the nozzle flow direction and 57 points in each dimension of the cross-flow plane. The number of grid points chosen allows the VULCAN-CFD solution to use a coarse-to-fine sequencing technique to a total of three levels (coarse, medium, fine). The grid was clustered near the nozzle throat due to the rapid changes in the nozzle profile shape in that region. The grid was also clustered near the walls in order to achieve y^+ values suitable for the use of a wall matching function.⁸

Convergence of the solution was achieved by local time stepping of the solution for 3000 iterations at each level of grid resolution. Figure D3 shows the

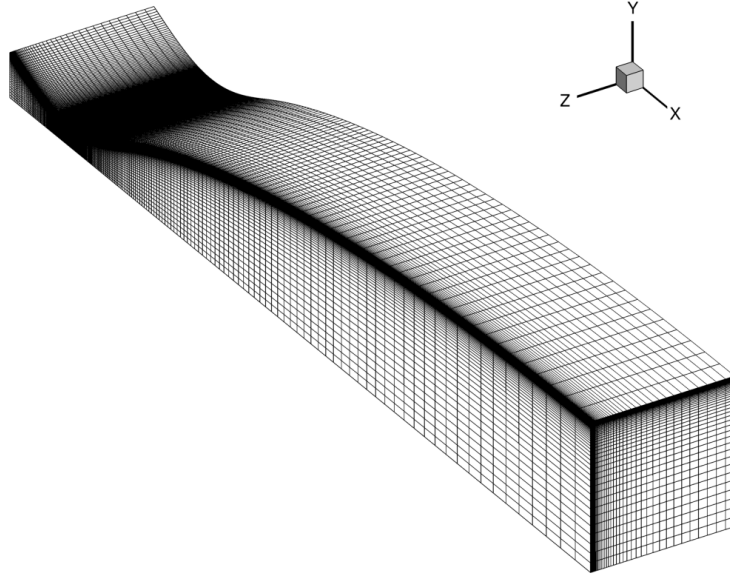


Figure D2. Nozzle grid for the quarter-domain is shown.

logarithmically-transformed residual, mass flow error, and CFL number as a function of iteration of the solution. Discontinuities in the mass flow error and residuals indicate points of grid refinement. The CFL number was also chosen to gradually increase to its maximum value (here chosen to be 25) over the coarse grid solution. Once converged, the y^+ values along the nozzle walls were verified to ensure they were suitable for solution using wall matching functions as shown in fig. D4. Because the maximum value was 16.42, it was determined that the grid resolution was suitable for the purposes of aerodynamic optimization.

The optimized nozzle profile and nozzle centerline longitudinal Mach number evolution are shown in fig. D5. The classical features of a supersonic nozzle are evident, with the flow beginning at subsonic conditions in the converging section and transitioning to supersonic conditions past the nozzle throat. The nozzle was also designed such that the nozzle profile would become parallel to the centerline at the nozzle exit. Note that due to boundary layer growth, having a parallel nozzle profile at the nozzle exit does not necessarily ensure that the flow is parallel to the centerline at all points across the nozzle exit plane.

Nozzle exit contours of Mach number and logarithmically-transformed pressure are shown in figs. D6 and D7, respectively. Both plots also feature a Mach contour line that represents 99% of the design Mach number in order to better visualize where the core is defined. While the entire region in excess of the 99% Mach number could be defined as the core, here it was defined simply as the box that fits within that contour as illustrated in fig. 2. The symmetry planes are defined by the location of the axes and the maxima of the axes define the adiabatic walls. In the Mach contour plot, the boundary layer is deformed into a “dog-bone” shape that is a result of the

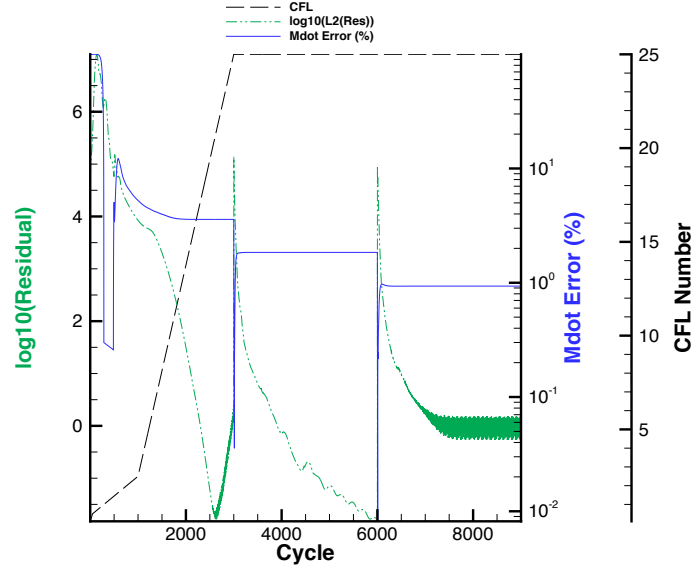


Figure D3. Typical errors, residuals, and CFL number evolution during a VULCAN-CFD nozzle flow simulation. Discontinuities indicate a refinement in the grid topology.

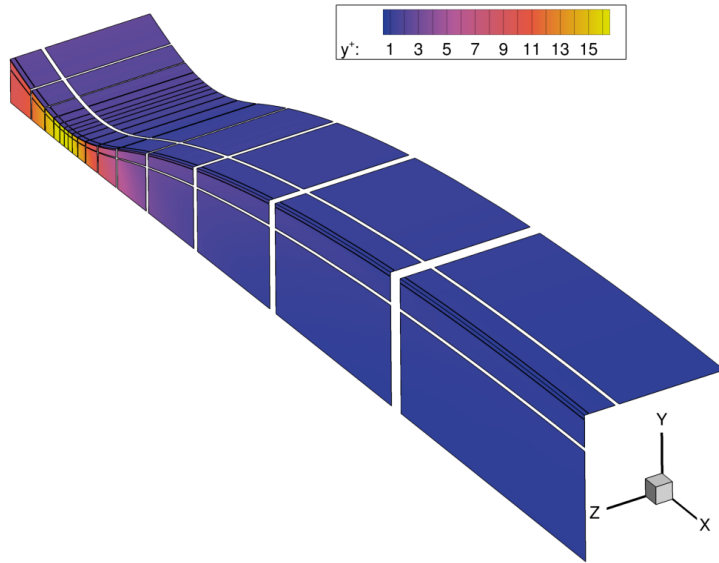


Figure D4. Representative y^+ for a solved case is shown. The maximum value of 16.42 occurs in the throat region of the flow.

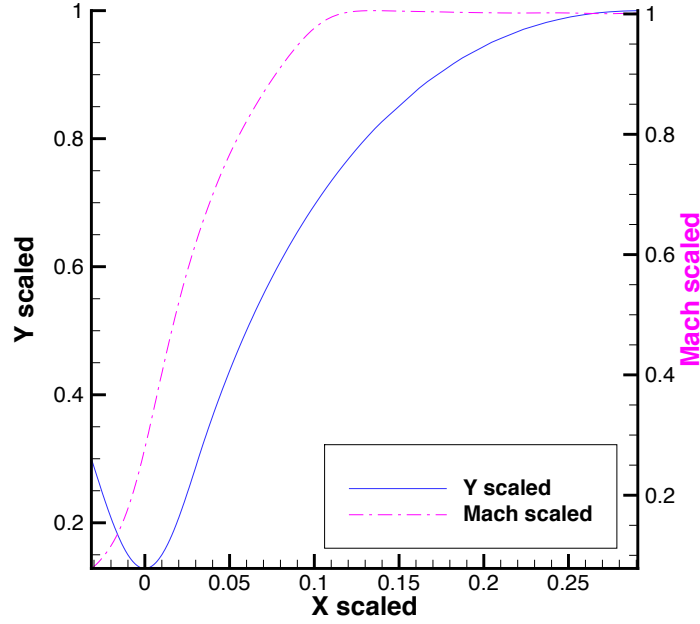


Figure D5. Scaled nozzle profile and evolution of centerline scaled Mach number for the length of the nozzle is shown. The Y-axis is scaled by nozzle exit height.

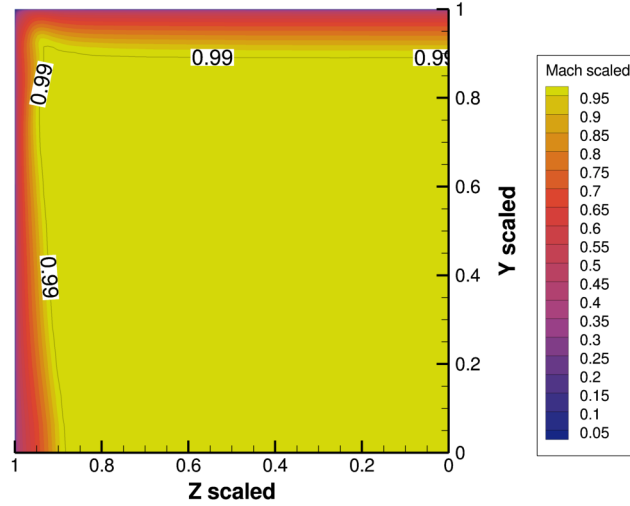


Figure D6. Scaled Mach contours in the nozzle core exit are shown for a quarter domain. The intersection of the symmetry planes is at the origin. The core is defined within the $\tilde{M} = 0.99$ contour, which is within 1% of the design Mach number. The Y- and Z-axis directions are both scaled by nozzle exit height.

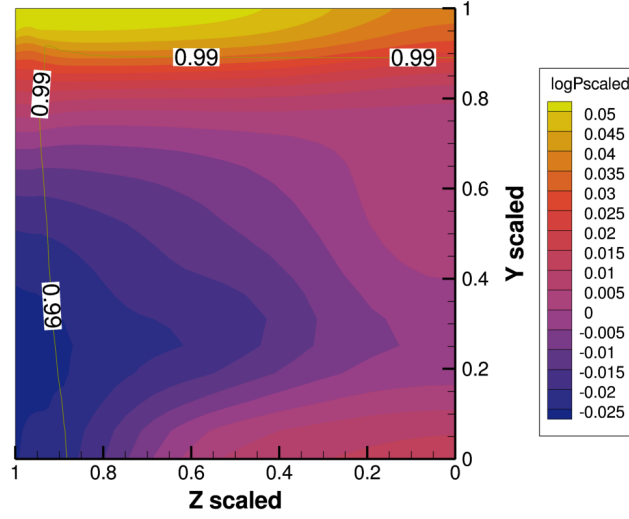


Figure D7. Logarithmically scaled pressures contours in the nozzle core exit are shown for a quarter domain. Pre-scaled pressures were non-dimensionalized by the mean nozzle exit static pressure. The intersection of the symmetry planes is at the origin. The core is defined within the $\tilde{M} = 0.99$ contour, which is within 1% of the design Mach number. The Y- and Z-axis directions are both scaled by nozzle exit height.

expansion of the nozzle wall in the Y-direction and the ensuing vorticity. Note that if expansion were allowed in the Z-direction as well this contour may have been further degraded in terms of flow uniformity and core size.²⁴

Due to the variation of the flow properties across the nozzle exit, uniformity of the flow properties was evaluated as a statistical measure using data points contained within the nozzle exit area. Table D1 summarizes an *a posteriori* statistical analysis of key nozzle parameters in the exit area. Furthermore, an average error for any given variable (denoted by the placeholder ϕ) was defined by

$$e_{\phi,avg} = \frac{\phi_{max} - \phi_{min}}{\phi_{mean}} \quad (D1)$$

that gave a metric to define the range of errors from the mean in the nozzle exit. These average errors for each tracked flow parameter are summarized in table D2. The results of tables D1 and D2 were for informational purposes only and were not fed back into the design process. However, a different optimization procedure could optimize on mean core Mach number instead of centerline Mach number and use the ensuing statistical parameters to constrain the allowable errors in the nozzle exit and use those values as constraints in the optimization algorithm.

Table D1. Statistical properties of nozzle core exit parameters. Mach number is scaled by the design value and all other properties are scaled by their mean value.

	\mathbf{M} [1]	$\tilde{\mathbf{p}}$ [1]	$\tilde{\mathbf{T}}$ [1]	$\tilde{\mathbf{p}}_0$ [1]
Mean	1.003	1.000	1.000	1.000
Standard Deviation	0.003	0.015	0.005	0.005
Min	0.992	0.975	0.993	0.965
Max	1.009	1.036	1.018	1.002

Table D2. Average error of nozzle exit parameters

	\mathbf{M}	\mathbf{p}	\mathbf{T}	\mathbf{p}_0
\mathbf{e}_{avg}	1.71%	6.04%	2.55%	3.76%

Appendix E

DAKOTA Iteration Data

The following subsections summarize the points sampled in the design space. Note that all function evaluations are not represented; only those that represent a candidate design point are shown. The column labeled **f1** represents the objective function value while **f2** represents the constraint value. **MachMOC** and **ThroatROC** are scaled MOC Mach number and nondimensionalized throat radius of curvature, respectively.

E.1 DIRECT Method

%eval	MachMOC	ThroatMOC	f1	f2
1	1.000	7.500	0.144	0.965
2	1.048	7.500	-0.237	1.001
3	0.952	7.500	0.142	0.919
4	1.048	9.167	-0.238	1.010
5	1.048	5.833	-0.247	1.010
6	1.063	5.833	-0.246	1.025
7	1.032	5.833	-0.241	0.997
8	1.000	9.167	0.149	0.965
9	1.000	5.833	0.138	0.965
10	1.053	5.833	-0.247	1.015
11	1.042	5.833	-0.247	1.005
12	1.048	9.722	-0.237	1.010
13	1.048	8.611	-0.234	1.003
14	0.968	7.500	0.142	0.934
15	0.937	7.500	0.141	0.904
16	1.042	6.389	-0.246	1.005
17	1.042	5.278	-0.250	1.005

18	1.063	7.500	-0.241	1.026
19	1.032	7.500	-0.237	0.996
20	0.968	9.167	0.147	0.935
21	0.968	5.833	0.137	0.935
22	1.044	5.278	-0.250	1.007
23	1.041	5.278	-0.250	1.003
24	1.048	6.389	-0.244	1.028
25	1.048	5.278	-0.250	1.010
26	1.069	5.833	-0.246	1.030
27	1.058	5.833	-0.246	1.020
28	0.952	9.167	0.147	0.921
29	0.952	5.833	0.137	0.921
30	1.041	5.278	-0.250	1.004
31	1.040	5.278	-0.250	1.003
32	1.053	6.389	-0.240	0.998
33	1.053	5.278	-0.250	1.015
34	1.037	5.833	-0.247	1.000
35	1.026	5.833	-0.247	1.001
36	0.937	9.167	0.146	0.905
37	0.937	5.833	0.136	0.906
38	1.040	5.278	-0.250	1.003
39	1.040	5.278	-0.250	1.003
40	1.037	6.389	-0.246	1.000
41	1.037	5.278	-0.250	1.000
42	1.069	7.500	-0.241	1.031
43	1.058	7.500	-0.242	1.021
44	1.000	6.389	0.142	0.965
45	1.000	5.278	0.137	0.965
46	1.039	5.278	-0.250	1.002
47	1.035	5.278	-0.250	0.998
48	1.026	6.389	-0.196	0.990
49	1.026	5.278	0.082	0.990
50	1.063	9.167	-0.237	1.025
51	1.032	9.167	-0.233	0.995
52	1.000	8.056	0.146	0.966
53	1.000	6.944	0.143	0.963
54	1.036	5.278	-0.250	0.999
55	1.035	5.278	-0.251	0.998
56	1.058	6.389	-0.244	1.026
57	1.058	5.278	-0.250	1.020
58	1.063	9.722	-0.236	1.025
59	1.032	9.722	-0.232	0.995
60	1.000	9.722	0.150	0.970
61	1.000	8.611	0.147	0.965
62	1.038	5.278	-0.250	1.001
63	1.036	5.278	-0.250	0.999

64	1.063	6.389	-0.244	1.025
65	1.063	5.278	-0.249	1.025
66	1.037	7.500	-0.237	1.001
67	1.026	7.500	0.118	0.991
68	1.039	5.463	0.136	0.956
69	1.039	5.093	-0.249	1.002
70	1.071	5.833	-0.246	1.031
71	1.067	5.833	-0.246	1.028
72	1.048	8.056	-0.241	1.010
73	1.048	6.944	-0.244	1.007

E.2 COBYLA Method

%eval	MachMOC	ThroatMOC	f1	f2
1	1.071	7.500	-0.241	1.034
2	1.357	7.500	-0.200	1.295
3	1.071	8.500	-0.239	1.032
4	1.033	6.509	-0.239	0.996
5	1.034	7.018	-0.238	0.994
6	1.039	6.769	-0.042	0.988
7	1.029	7.267	-0.214	0.991
8	1.036	7.142	-0.238	0.997
9	1.016	7.013	0.144	0.977
10	1.035	7.143	-0.238	0.995
11	1.040	6.960	-0.244	1.000
12	1.040	6.991	-0.244	1.000
13	1.040	7.023	-0.243	1.012
14	1.042	6.978	-0.244	1.002
15	1.042	6.977	-0.244	1.002
16	1.041	6.984	-0.244	1.001
17	1.039	6.990	-0.244	0.999
18	1.040	6.999	-0.244	1.000
19	1.040	7.007	-0.244	1.000
20	1.040	7.015	-0.243	1.012
21	1.039	7.007	-0.245	0.999
22	1.040	7.015	-0.243	1.012
23	1.040	7.011	-0.244	1.000
24	1.040	7.015	-0.243	1.012
25	1.040	7.013	-0.245	1.000
26	1.040	7.015	-0.243	1.012
27	1.041	7.013	-0.244	1.000
28	1.041	7.013	-0.243	1.013
29	1.040	7.012	-0.244	1.000
30	1.040	7.013	-0.244	1.000
31	1.040	7.013	-0.245	1.000
32	1.040	7.013	-0.245	1.000

33	1.040	7.014	-0.245	1.000
34	1.040	7.014	-0.243	1.012
35	1.040	7.014	-0.245	1.000
36	1.040	7.014	-0.243	1.012
37	1.040	7.014	-0.245	1.000
38	1.040	7.014	-0.245	1.000
39	1.040	7.014	-0.245	1.000
40	1.040	7.014	-0.245	1.000
41	1.040	7.014	-0.245	1.000
42	1.040	7.014	-0.245	1.000
43	1.040	7.014	-0.245	1.000
44	1.040	7.014	-0.243	1.012

E.3 Quasi-Newton Method

%eval	MachMOC	ThroatMOC	f1	f2
1	1.071	5.000	-0.253	1.032
2	1.042	7.105	-0.244	1.002
3	1.020	9.433	0.150	0.984
4	1.031	8.269	-0.228	0.994
5	1.036	7.687	-0.236	1.001
6	1.039	7.396	-0.243	1.002
7	1.040	7.251	-0.244	1.002

REPORT DOCUMENTATION PAGE					Form Approved OMB No. 0704-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>						
1. REPORT DATE (DD-MM-YYYY) 01-08-2015		2. REPORT TYPE Technical Memorandum		3. DATES COVERED (From - To) 10/2013-9/2014		
4. TITLE AND SUBTITLE An Automated DAKOTA and VULCAN-CFD Framework with Application to Supersonic Facility Nozzle Flowpath Optimization				5a. CONTRACT NUMBER		
				5b. GRANT NUMBER		
				5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Axdahl, Erik L.				5d. PROJECT NUMBER		
				5e. TASK NUMBER		
				5f. WORK UNIT NUMBER 110076.02.07.05		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Langley Research Center Hampton, Virginia 23681-2199				8. PERFORMING ORGANIZATION REPORT NUMBER L-20571		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001				10. SPONSOR/MONITOR'S ACRONYM(S) NASA		
				11. SPONSOR/MONITOR'S REPORT NUMBER(S) NASA/TM-2015-218796		
12. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified-Unlimited Subject Category 66 Availability: NASA STI Program (757) 864-9658						
13. SUPPLEMENTARY NOTES An electronic version can be found at http://ntrs.nasa.gov .						
14. ABSTRACT Removing human interaction from design processes by using automation may lead to gains in both productivity and design precision. This memorandum describes efforts to incorporate high fidelity numerical analysis tools into an automated framework and applying that framework to applications of practical interest. The purpose of this effort was to integrate VULCAN-CFD into an automated, DAKOTA-enabled framework with a proof-of-concept application being the optimization of supersonic test facility nozzles. It was shown that the optimization framework could be deployed on a high performance computing cluster with the flow of information handled effectively to guide the optimization process. Furthermore, the application of the framework to supersonic test facility nozzle flowpath design and optimization was demonstrated using multiple optimization algorithms.						
15. SUBJECT TERMS DAKOTA; VULCAN-CFD; optimization; automation; nozzle						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			STI Information Desk (email: help@sti.nasa.gov)	
U	U	U	UU	36	19b. TELEPHONE NUMBER (Include area code) (757) 864-9658	